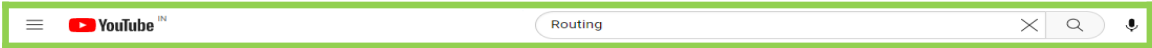
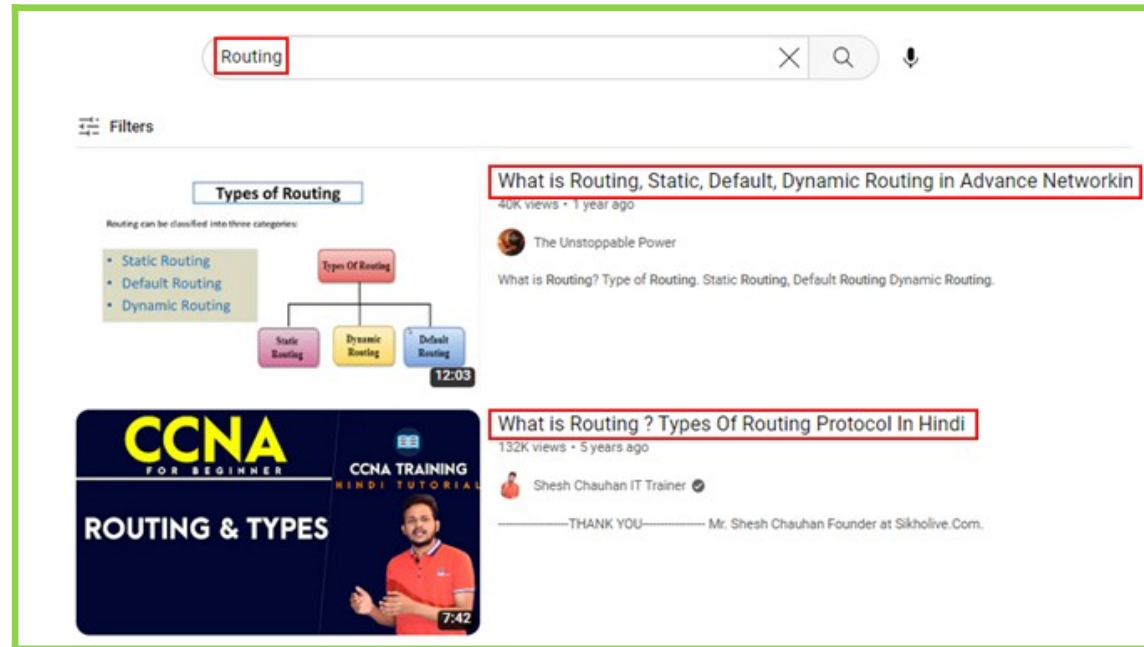


# Exhibit 7.2

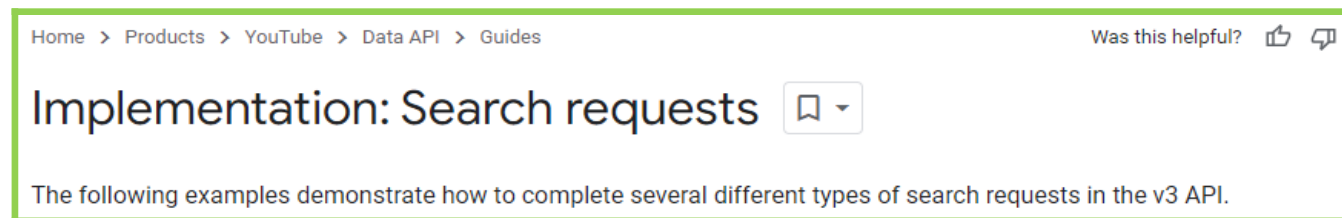
## Infringement Claim Chart for U.S. Pat. No. US10237420B1 v. Youtube ("Defendant")

Claims	Evidence
20. A method of processing requests, comprising:	<p>The Youtube platform with system-generated Searching Service performs a method of processing requests. For Example, Youtube Searching Service receive user request, process them, and provides needed information right at the system. So, it helps you find what you're looking for and stay in control of what you see.</p> <div data-bbox="514 548 1923 669" style="border: 1px solid green; padding: 10px;"> <p>Our goal is to give you the resources to help you understand factors that we take into account to provide the best search results for users.</p> </div> <p>Source: <a href="https://www.youtube.com/intl/ALL_in/howyoutubeworks/product-features/search/#trusted-results">https://www.youtube.com/intl/ALL_in/howyoutubeworks/product-features/search/#trusted-results</a></p> <div data-bbox="543 818 1894 967" style="border: 1px solid green; padding: 10px;"> <p>Returns a collection of search results that match the query parameters specified in the API request. By default, a search result set identifies matching <code>video</code>, <code>channel</code>, and <code>playlist</code> resources, but you can also configure queries to only retrieve a specific type of resource. <a href="#">Try it now.</a></p> </div> <p>Source: <a href="https://developers.google.com/youtube/v3/docs/search">https://developers.google.com/youtube/v3/docs/search</a></p> <div data-bbox="518 1081 1919 1208" style="border: 1px solid green; padding: 10px;"> <p>For clarity of presentation, <u>the samples on this page show the distinctive elements of each request and abbreviate the base URL for the host that processes Data API requests ( <a href="https://www.googleapis.com/youtube/v3">https://www.googleapis.com/youtube/v3</a> ).</u> To make the request outside of the context of the samples, you need to include the full URL.</p> </div> <p>Source: <a href="https://developers.google.com/youtube/v3/sample_requests">https://developers.google.com/youtube/v3/sample_requests</a></p> <div data-bbox="646 1320 1791 1367" style="border: 1px solid green; padding: 5px;">  </div>

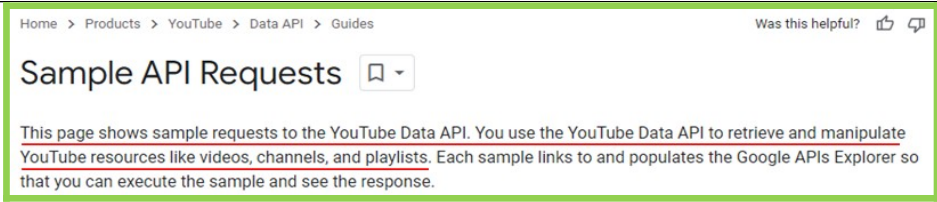
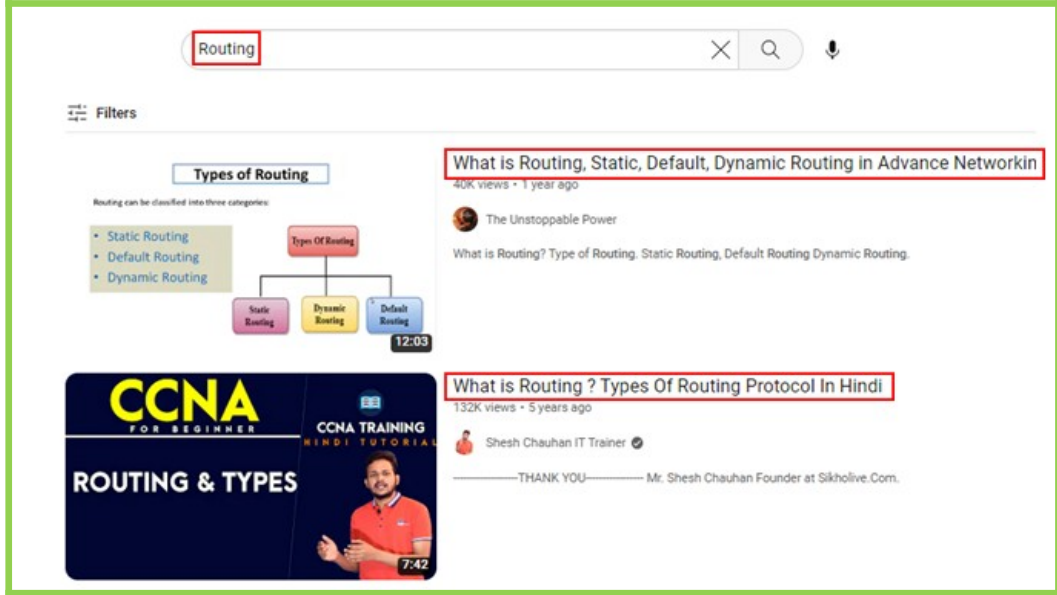
Source: [https://www.youtube.com/results?search\\_query=Routing](https://www.youtube.com/results?search_query=Routing)



Source: [https://www.youtube.com/results?search\\_query=Routing](https://www.youtube.com/results?search_query=Routing)

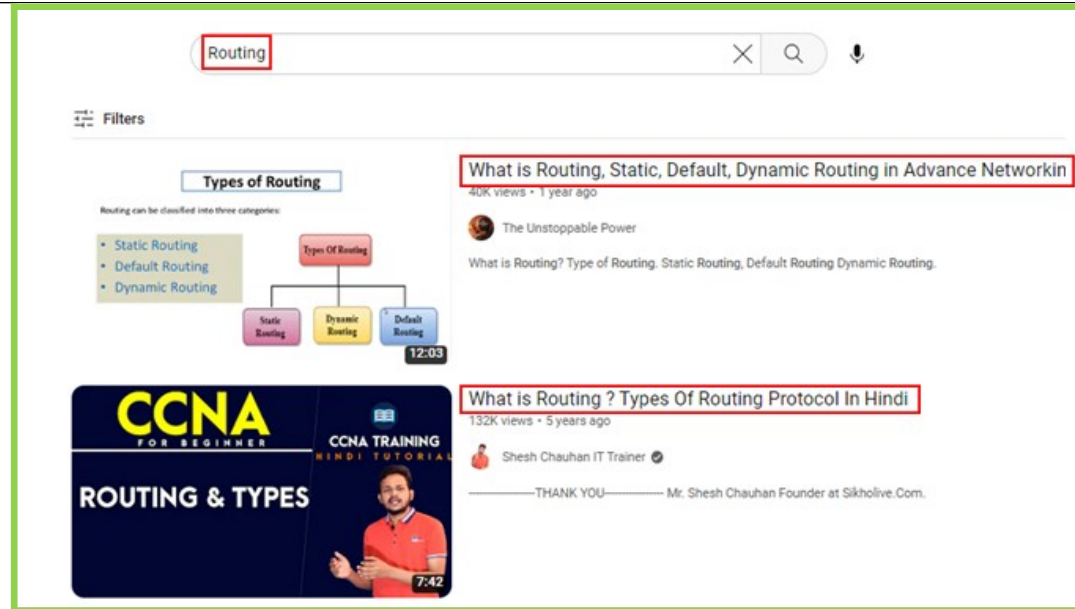


Source: <https://developers.google.com/youtube/v3/guides/implementation/search>

	 <p>Source: <a href="https://developers.google.com/youtube/v3/sample_requests">https://developers.google.com/youtube/v3/sample_requests</a></p>
<p>estimating at least one content-specific or requestor-specific characteristic associated with each received request;</p>	<p>The Youtube platform with system-generated Searching Service estimates at least one content-specific or requestor-specific characteristic associated with each received request. For Example, Youtube estimates at least one content-specific (i.e., search query parameters (what the user looking for)) or requestor-specific (user's intent) characteristic associated with each received request.</p>  <p>Source: <a href="https://www.youtube.com/results?search_query=Routing">https://www.youtube.com/results?search_query=Routing</a></p>

	<div data-bbox="604 240 737 276"> <h3><u>Playlists</u></h3> </div> <div data-bbox="634 313 1820 375"> <p>This example shows how to find playlists matching the query term "GoogleDevelopers." It calls the <code>search.list</code> method and sets the <code>type</code> parameter's value to <code>playlist</code> so that the result set only includes playlists.</p> </div> <p>Source: <a href="https://developers.google.com/youtube/v3/guides/implementation/search">https://developers.google.com/youtube/v3/guides/implementation/search</a></p> <div data-bbox="625 532 741 568"> <h3><u>Videos</u></h3> </div> <div data-bbox="659 605 1799 662"> <p>This example calls the <code>search.list</code> method to find the most viewed, high-definition (HD) videos associated with the query "skateboarding dog." The query sets the <code>order</code>, <code>part</code>, <code>q</code>, <code>type</code>, and <code>videoDefinition</code> parameters.</p> </div> <p>Source: <a href="https://developers.google.com/youtube/v3/guides/implementation/search">https://developers.google.com/youtube/v3/guides/implementation/search</a></p> <div data-bbox="571 821 783 857"> <h3><u>Channels</u> ⇄</h3> </div> <div data-bbox="609 899 1856 961"> <p>This example shows how to find channels matching the query term "travel." It calls the <code>search.list</code> method and sets the <code>type</code> parameter's value to <code>channel</code> so that the result set only includes channels.</p> </div> <p>Source: <a href="https://developers.google.com/youtube/v3/guides/implementation/search">https://developers.google.com/youtube/v3/guides/implementation/search</a></p>
<p>determining availability of a plurality of alternate target resources, each respective target resource</p>	<p>The Youtube platform with system-generated Searching Service determines a set of available alternate target resources, each having at least one respective target characteristic. For Example, YouTube uses natural language understanding technologies to understand what is being requested and on the basis of this, determine resources (content) on the characteristic such as current availability and query matching of the resources.</p>

having at least one respective target characteristic;



Source: [https://www.youtube.com/results?search\\_query=Routing](https://www.youtube.com/results?search_query=Routing)

A search result contains information about a YouTube video, channel, or playlist that matches the search parameters specified in an API request. While a search result points to a uniquely identifiable resource, like a video, it does not have its own persistent data.

Source: <https://developers.google.com/youtube/v3/docs/search>

id	object
	The <code>id</code> object contains information that can be used to uniquely identify the resource that matches the search request.

Source: <https://developers.google.com/youtube/v3/docs/search>

`id.videoId``string`

If the `id.type` property's value is `youtube#video`, then this property will be present and its value will contain the ID that YouTube uses to uniquely identify a video that matches the search query.

Source: <https://developers.google.com/youtube/v3/docs/search>

`id.channelId``string`

If the `id.type` property's value is `youtube#channel`, then this property will be present and its value will contain the ID that YouTube uses to uniquely identify a channel that matches the search query.

Source: <https://developers.google.com/youtube/v3/docs/search>

The response to each request is the `JSON` representation of a YouTube resource. The `part` parameter in the request specifies which portions of the resource are included in the response. The parameter identifies one or more top-level (non-nested) resource properties that should be included in the response. For example, some of the parts of a `video` resource are:

- snippet
- contentDetails
- player
- statistics
- status

Source: [https://developers.google.com/youtube/v3/sample\\_requests](https://developers.google.com/youtube/v3/sample_requests)

evaluating,  
with at least

The Youtube platform with system-generated Searching Service evaluates, with the automated processor, a plurality of alternate allocations of the respective received



one automated processor, a plurality of alternate allocations of the respective received request with different available targets, according to a ranking dependent on a probabilistic predictive multivariate evaluator, based on the at least one content-specific or requestor-specific characteristic, and the respective target characteristics of the plurality of alternate target resources; and

request with different available targets, according to a ranking dependent on a probabilistic predictive multivariate evaluator, based on the at least one content-specific or requestor-specific characteristic, and the respective target characteristics of the plurality of alternate target resources.

For Example, YouTube uses the content-specific or requestor-specific characteristics of the request and the availability and characteristics parameters of the target resources to evaluate a plurality of alternate allocations of the respective received request with different available resources via artificial intelligence techniques such as neural networks and machine learning.

With over 500 hours of content uploaded to YouTube every minute, finding what you need would be nearly impossible without some help sorting through all the videos. YouTube's search ranking system does just that by sorting through a vast number of videos to find the most relevant and useful results to your search query and presenting them in a way that helps you find what you're looking for.

Source: [https://www.youtube.com/intl/ALL\\_in/howyoutubeworks/product-features/search/](https://www.youtube.com/intl/ALL_in/howyoutubeworks/product-features/search/)

#### **Application of Artificial Intelligence by YouTube**

Below are a couple of ways through which YouTube's platform adopts artificial intelligence in the present day

In response to this issue, the solution the platform churned up is basically a two-part system. In this system the first part would be the candidate generation, in which the algorithm examines the **history of the user on the platform**. The second part is the ranking system. This system accredits a score to every video.

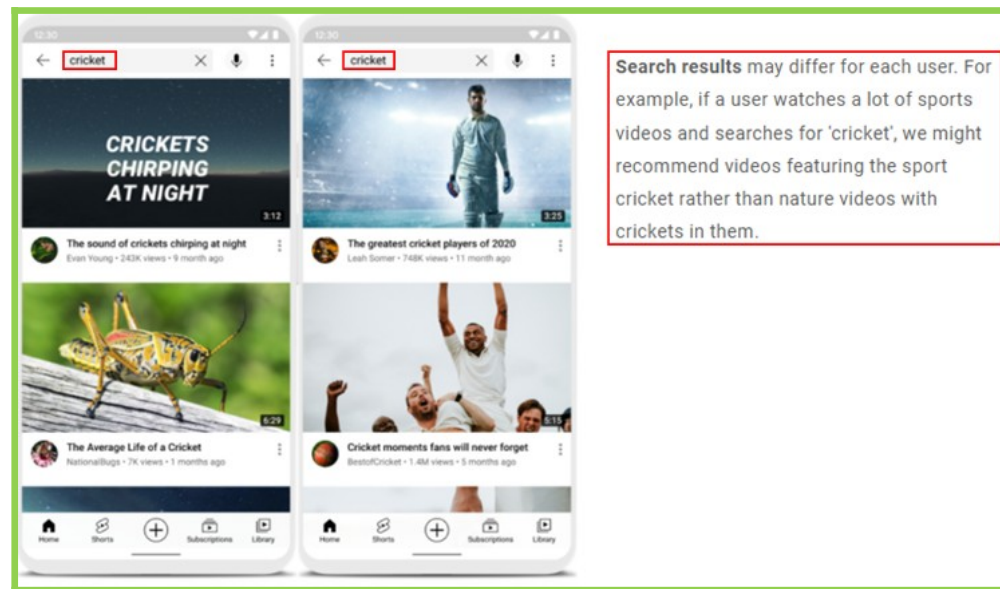


Source: <https://www.analyticssteps.com/blogs/how-youtube-using-artificial-intelligence>

The primary aim here is not to determine “good” videos but rather to match the users with the videos which they wish to watch in order to ensure that they spend a maximum amount of time on the platform.

Presently the working of the YouTube recommendation system is basically along the lines of the following. In layman’s terms, in order to fill the sidebar with recommended videos, the platform firstly assembles a shortlist consisting of over hundred videos by determining the videos which complement the topic and the various other characteristics of the video the user is presently streaming. Following this, the platform arranges a list, ranking it as per the preference of the user, which it absorbs by supplying the user’s clicks, likes and remaining interactions into a machine learning algorithm.

Source: <https://www.analyticssteps.com/blogs/how-youtube-using-artificial-intelligence>



Source: [https://www.youtube.com/intl/ALL\\_in/howyoutubeworks/product-features/search/](https://www.youtube.com/intl/ALL_in/howyoutubeworks/product-features/search/)

## Documentation ► youtube.search.list

Returns a collection of search results that match the query parameters specified in the API request. By default, a search result set identifies matching video, channel, and playlist resources, but you can also configure queries to only retrieve a specific type of resource.

Source: [https://www.any-api.com/googleapis\\_com/youtube/docs/search/youtube\\_search\\_list](https://www.any-api.com/googleapis_com/youtube/docs/search/youtube_search_list)

**channelId:string**

The channelId parameter indicates that the API response should only contain resources created by the channel

Source: [https://www.any-api.com/googleapis\\_com/youtube/docs/search/youtube\\_search\\_list](https://www.any-api.com/googleapis_com/youtube/docs/search/youtube_search_list)

**channelType:string**

The channelType parameter lets you restrict a search to a particular type of channel.

Allowed values are:

- any
- show

Source: [https://www.any-api.com/googleapis\\_com/youtube/docs/search/youtube\\_search\\_list](https://www.any-api.com/googleapis_com/youtube/docs/search/youtube_search_list)

**eventType:string**

The eventType parameter restricts a search to broadcast events. If you specify a value for this parameter, you must also set the type parameter's value to video.

Allowed values are:

- completed
- live
- upcoming

Source: [https://www.any-api.com/googleapis\\_com/youtube/docs/search/youtube\\_search\\_list](https://www.any-api.com/googleapis_com/youtube/docs/search/youtube_search_list)

	<div data-bbox="743 191 1692 313" style="border: 1px solid green; padding: 5px;"> <p><code>maxResults:integer</code>  The maxResults parameter specifies the maximum number of items that should be returned in the result set.</p> </div> <p>Source: <a href="https://www.any-api.com/googleapis_com/youtube/docs/search/youtube_search_list">https://www.any-api.com/googleapis_com/youtube/docs/search/youtube_search_list</a></p> <div data-bbox="816 423 1619 740" style="border: 1px solid green; padding: 5px;"> <p><code>order:string</code>  The order parameter specifies the method that will be used to order resources in the API response.  Allowed values are:</p> <ul style="list-style-type: none"> <li>• date</li> <li>• rating</li> <li>• relevance</li> <li>• title</li> <li>• videoCount</li> <li>• viewCount</li> </ul> </div> <p>Source: <a href="https://www.any-api.com/googleapis_com/youtube/docs/search/youtube_search_list">https://www.any-api.com/googleapis_com/youtube/docs/search/youtube_search_list</a></p>
<p>generating a control signal , by the at least one automated processor, selectively dependent on the evaluating, to control the allocations of the respective received request with the different</p>	<p>The Youtube platform with system-generated Searching Service generates a control signal, by the automated processor, selectively dependent on the evaluating, to control the allocations of the respective received request with the different available targets. For Example, responsive to the evaluation, Youtube generates a control signal for the allocation of the different available resources. The control signal is selectively dependent on the evaluation in view of other factors such as the overall throughput of the system and the priority and requirements of other concurrent requests.</p>

available targets.

Services often support multiple access scopes. Each scope specifies the resources that an API Client can retrieve, insert, update, or delete on the user's behalf. Scopes enable API Clients to only request access to the resources they need, and scopes also enable users to control the amount of access that they grant to those Clients.

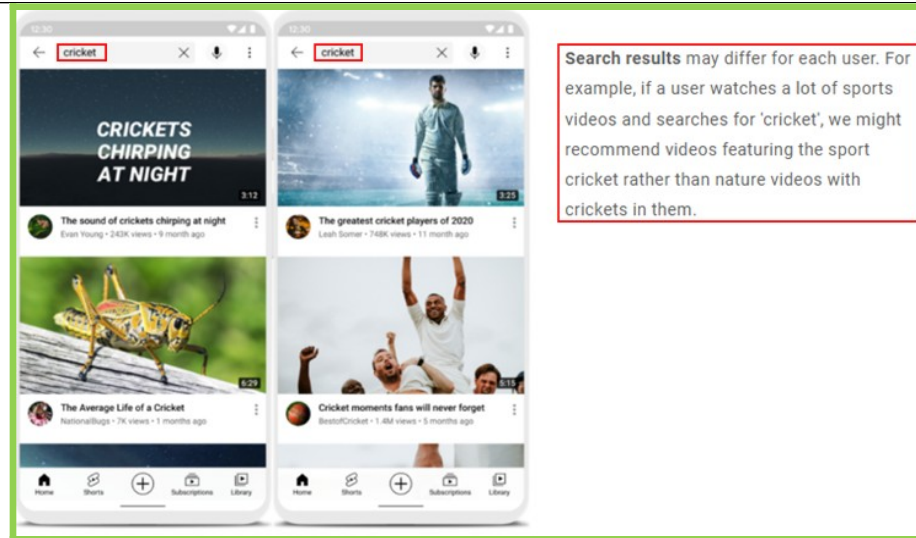
Source: <https://developers.google.com/youtube/terms/developer-policies>

Returns a collection of search results that match the query parameters specified in the API request. By default, a search result set identifies matching `video`, `channel`, and `playlist` resources, but you can also configure queries to only retrieve a specific type of resource. [Try it now](#).

Source: <https://developers.google.com/youtube/v3/docs/search>

At YouTube Search, we prioritise three main elements to provide the best search results: relevance, engagement and quality. These three elements are given differing importance based on the type of search. To estimate relevance we look into many factors, such as how well the title, tags, description and video content match your search query. Engagement signals are a valuable way to determine relevance. We incorporate aggregate engagement signals from users, i.e. we may look at the watch time of a particular video for a particular query to determine if the video is considered relevant to the query by other users. Finally, for quality, our systems are designed to identify signals that can help determine which channels demonstrate expertise, authoritativeness and trustworthiness on a given topic. YouTube doesn't accept payment for better placement within organic search results.

Source: [https://www.youtube.com/intl/ALL\\_in/howyoutubeworks/product-features/search/](https://www.youtube.com/intl/ALL_in/howyoutubeworks/product-features/search/)



Source: [https://www.youtube.com/intl/ALL\\_in/howyoutubeworks/product-features/search/](https://www.youtube.com/intl/ALL_in/howyoutubeworks/product-features/search/)

The primary aim here is not to determine “good” videos but rather to match the users with the videos which they wish to watch in order to ensure that they spend a maximum amount of time on the platform.

Presently the working of the YouTube recommendation system is basically along the lines of the following. In layman’s terms, in order to fill the sidebar with recommended videos, the platform firstly assembles a shortlist consisting of over hundred videos by determining the videos which complement the topic and the various other characteristics of the video the user is presently streaming. Following this, the platform arranges a list, ranking it as per the preference of the user, which it absorbs by supplying the user’s clicks, likes and remaining interactions into a machine learning algorithm.

Source: <https://www.analyticssteps.com/blogs/how-youtube-using-artificial-intelligence>